Subject: Comments on Section 4
From: Van Snyder

1 Introduction

The Editor remarked that he liked the introduction in 02-167r1 better than its edits for 4.5.1.6. The discussion presently in 4.5.1.6 doesn't address procedure pointer components with implicit interface, and is defective in other ways.

2 Technical changes

(1) Subgroup concluded that default PASS for procedure pointer components and named type-bound procedures is better than expecting a human reader (of code or standard) to deduce whether PASS or NOPASS is implied. This paper proposes that PASS is the default if the first dummy argument satisfies the requirement to be a passed-object dummy argument. Otherwise, either PASS or NOPASS shall be specified. I.e., instead of implicit determination of PASS or NOPASS, PASS is the default under strictly circumscribed conditions.

(2) For type-bound operators, assignment or user-defined derived-type input/output, there is no *part-ref* from which to determine the dynamic type to use for dynamic dispatch. As such, constraint C458 at [45:20] is incorrect and incomplete: it needs to *require* PASS for all three instead of *prohibiting* it for the *dtio-generic-spec* case.

(3) As a consequence of (2), the `unit` and `dtv` arguments of user-defined derived-type input/output procedures are interchanged.

3 Edits

Edits refer to 02-007r2. Page and line numbers are displayed in the margin. Absent other instructions, a page and line number or line number range implies all of the indicated text is to be replaced by associated text, while a page and line number followed by + (-) indicates that associated text is to be inserted after (before) the indicated line. Remarks are noted in the margin, or appear between [and] in the text.

[Editor: Delete because C463$\frac{1}{2}$ (see edit for 51:8-18 below) covers it. Then insert the following in its place:] 43:31-44:5

[TECHNICAL CHANGE 1:]

C438 (R436) If the procedure pointer component has an implicit interface or has no arguments, NOPASS shall be specified.

C439 (R436) If PASS (*arg-name*) appears, the interface shall have a dummy argument named *arg-name*.

[C440 could be construed to prohibit PASS and NOPASS from appearing together in the same type definition, or even in the same scoping unit. Make sure it applies to one *proc-component-attr-spec-list* at a time. Editor: Insert "same" after "the".] 44:4

[Editor: Delete because C463$\frac{1}{2}$ (see edit for 51:8-18 below) covers it. Then insert the following in its place:] 45:11-18

[TECHNICAL CHANGE 1:]

C455 (R441, R442) If the interface of the binding has no dummy argument of the type being defined, NOPASS shall appear.

C456 (R441, R442) If PASS (*arg-name*) appears, the interface of the binding shall have a dummy argument named *arg-name*.

[TECHNICAL CHANGE 2:] 45:20

C457 (R442) A *generic-binding* for which *generic-spec* is not *generic-name* shall have a passed-object dummy argument (4.5.1.6).

A **passed-object dummy argument** is a distinguished dummy argument of a procedure pointer component or type-bound procedure. It affects procedure overriding (4.5.3.2) and argument association (12.4.1.1). 51:8-18

If NOPASS is specified, the procedure pointer component or type-bound procedure has no passed-object dummy argument.

[TECHNICAL CHANGE 1:]

If neither PASS nor NOPASS is specified or PASS is specified without *arg-name*, the first dummy argument of a procedure pointer component or type-bound procedure is its passed-object dummy argument.

If PASS (*arg-name*) is specified, the dummy argument named *arg-name* is the passed-object dummy argument of the procedure pointer component or named type-bound procedure.

C463$\frac{1}{2}$ The passed-object dummy argument shall be a scalar, nonpointer, nonallocatable dummy data object with the same declared type as the type being defined; all of its nonkind type parameters shall be assumed; it shall be polymorphic if and only if the type being defined is extensible.

[C461 allows external procedures. Editor: Delete "module" from the first line of Note 4.33.] 51:18+2

[TECHNICAL CHANGE 3:]

To simplify compliance with C457 (see edit for [45:20]), make the `dtv` argument the first one.

Editor: Exchange the `unit` and `dtv` arguments at 201:20-21, 201:36-37, 202:4-5 and 202:20-21. Move the declaration of the `unit` arguments to be after the `dtv` arguments at 201:24 → 201:26+, 201:39 → 201:41+, 202:8 → 201:10+ and 202:23 → 202:25+.]